



# The Building in Oct 2001

---





# A tour of Cambridge

---

Derek McAuley  
Deputy Director  
MSR Cambridge



## A tour of Cambridge

---

Derek McAuley  
Deputy Director  
MSR Cambridge



## Cambridge Lab Overview

---

- Lab established in June, 1997
- Pan-European
  - msrctfrog has 11 members!
  - 12 nationalities
- Now at 50 researchers
- Close collaboration with Cambridge University
  - Computer Lab for language, security and systems
  - Engineering for machine learning and vision
  - Stats Lab for networking



## MSR / CU interactions

---

- Joint projects
- Co-supervision of PhD students
- Several lecture courses given
- Drink in the same pub
- Open seminars



## A typical month

---

|           |                |   |     |
|-----------|----------------|---|-----|
| 05-Jun-00 | Tom Ball       | Boolean Programs: A Model and Process for Software        | EXT |
| 05-Jun-00 | Rick Rashid    | The Future – it isn't what it used to be.                 | INT |
| 07-Jun-00 | Peter Druschel | The ScalaServer project: Designing scalable, high-        | EXT |
| 08-Jun-00 | Steve Muir     | Operating System Support for High-Performance Server      | EXT |
| 08-Jun-00 | Alex Smola     | MLP Sparse Greedy Methods for Learning                    | EXT |
| 09-Jun-00 | Hugo Zaragoza  | A Dynamic Probability Model from Complex Information...   | EXT |
| 13-Jun-00 | Tuomas Aura    | Extending the applications of public-key certificates     | EXT |
| 19-Jun-00 | Arno Puder     | MICO- an Open Source CORBA implementation                 | EXT |
| 20-Jun-00 | Brandan Frey   | MLP Learning mixtures of smooth, nonuniform deformation   | EXT |
| 21-Jun-00 | Mauro Pezze    | Testing Object Oriented Software                          | EXT |
| 29-Jun-00 | Patrick Dussud | The COM+ EE Garbage Collector                             | INT |
| 30-Jun-00 | Adrian Perrig  | Efficient authentication and signing of multicast streams | EXT |



## Our subjects

---

- Computer Security
- Information Retrieval And Analysis
- Machine Learning and Perception
- Networking
- Systems and Performance
- Programming Principles and Tools



## Security

---

- Theory and language for security and mobility
- Security support for product
- http-SIM
  - Implementation of http server on WfSC
  - "Access enabling wallets on user controlled devices"





# Computational Number Theory

- GIMPS (Great Internet Mersenne Prime Search)
  - Mersenne numbers  $M_p = 2^p - 1$
  - 1999 result  $M_{6972593}$  is prime
  - Why? Because it's there...
- CABAL (~20 members include MSR (CSR), CWI, Citibank, Sun, INRIA,...)
  - Last year broke RSA-155 challenge
  - Why? Because 512 bit RSA keys protect the global financial systems....
  - Seeking parallelization of linear algebra



## "Traditional" machine learning

---

- The usual suspects:
  - Speech
  - Handwriting
  - Image content
  - Textual semantics
  - ...



## "Relevance Vector Machine"

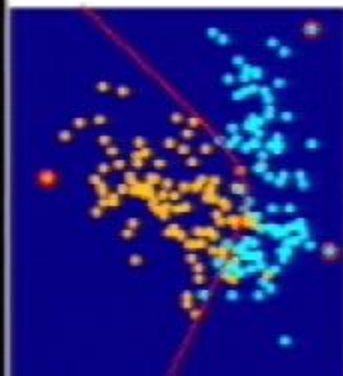
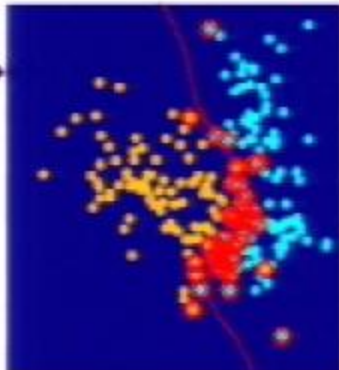
---

- Provides high recognition accuracy
- Key feature: very compact
  - runs very quickly
  - requires minimal memory storage
- e.g. handwriting in a PDA....



## RVM Illustration

The state-of-the-art  
"support" vector machine (right)  
uses 44 vectors (out of 200)



◀ The relevance vector machine (left)  
requires only 3 vectors



# LiveSpline Project

---

Cut and paste





## LiveSpline Project



Cut and paste





## LiveWire v. LiveSpline

---

- Pixel contour
- Background leakage
- Aliasing

V.

- Curve as contour (in future learn priors)
- Subpixel edge detection
- Fix alpha computation



# LiveSpline



Composited image



LiveWire



LiveSpline





## New MSR building

---



- Ground-breaking on **11th July 2000.**
- Completion next year.
- We want reconstruct a **3D Virtual model** of the building while it is being constructed.



## Cambridge from the air

---





## Cambridge from the air

---





## Cambridge from the air

---





# Cambridge from the air

---





# The Building in Oct 2001

---





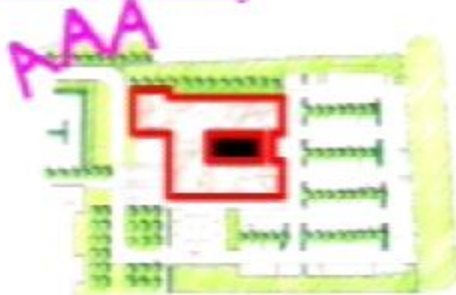
# Cambridge from the air

---



# Virtual MSRC project

Cameras looking at a corner of the building



- Three cameras are being installed next to the building site.
- Process image with Computer Vision tools.
- Construct 3D model and put on the web.





## Today, There's a crane

---



Image from 09-Aug-2000

- Developing the necessary **theory** and **software**.
- Testing on example buildings
- 3 (uncalibrated) images of **small resolution** (640x480)...

# West Cambridge



UNIVERSITY OF  
CAMBRIDGE

Web-based Participation for Campus Scale Project Design: The West Cambridge Site

The Site  
The Masterplan  
Overview  
Webcams  
Site Users  
Site Buildings  
Transportation  
Context  
What's New  
Web Consultation  
About this website  
Contact Us

## webcams



Camera 1



Camera 2

## Fast Links

[Cameras 1-10](#)  
[Project Site](#)

## Camera Facts

Camera Resolution 768 x 576





## An example building



The three input images



Images of the reconstructed 3D model





## Future Work

---

Improving the **reconstruction details** and **reducing artefacts** by:

- better matching of features across views;
- improving the reconstruction of edges;
- better occlusion reasoning;
- plane fitting to planar walls.

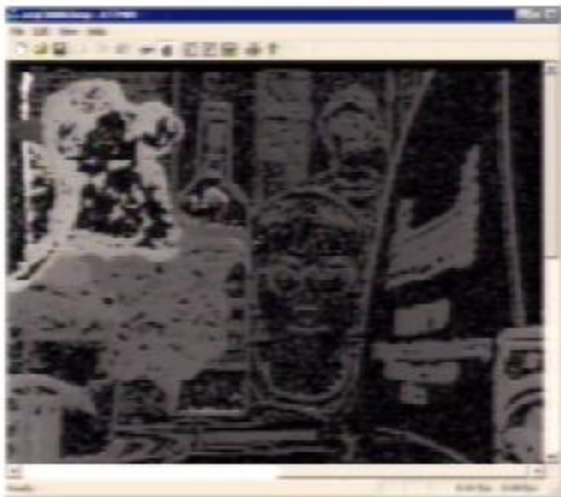


# HocusFocus





# HocusFocus index







## HocusFocus Result



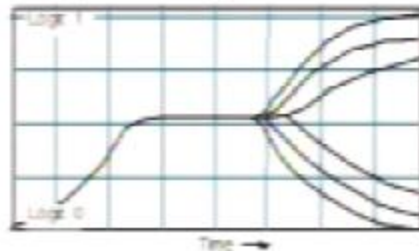


# Hardware fails

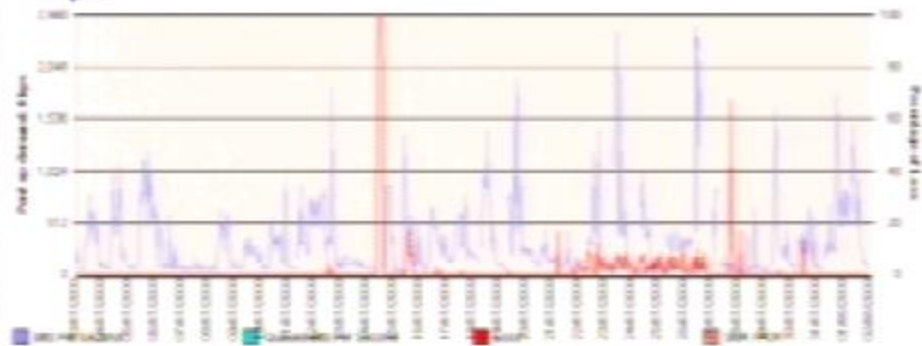
- ...a philosophical problem since 1328

**Buridan, Jean** (1300–1360), French Scholastic philosopher, who held a theory of **indifference**, contending that one will must choose the greater good, must it exclude. Buridan was educated at the University of Paris, where he studied with the English Scholastic philosopher William of Ockham. After his studies were completed, he was appointed professor of **philosophy** and later Rector at the same university. Buridan is traditionally, but probably incorrectly, associated with a philosophical dilemma of moral choice called "Buridan's ass," in the problem an **ass** starved to death between two equally alluring and equidistant bundles of hay because it has no rational basis for preferring one bundle over the other.

- It's not just bad engineering



# Networks drop packets



■ Cambridge MSR / London link

# Networks drop packets



- Cambridge MSR / London link



## Consider a well known game...

---



- A board of 41 squares
- Moves based on the roll of two dice
- Some rules...



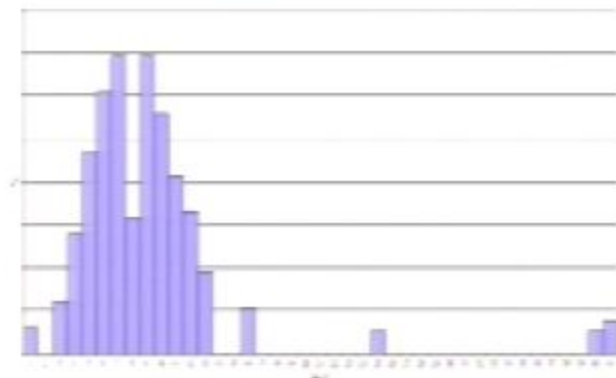
## Learn by Observation

- We watch some games and learn
- The probability of being on a particular square:





## Learning more...

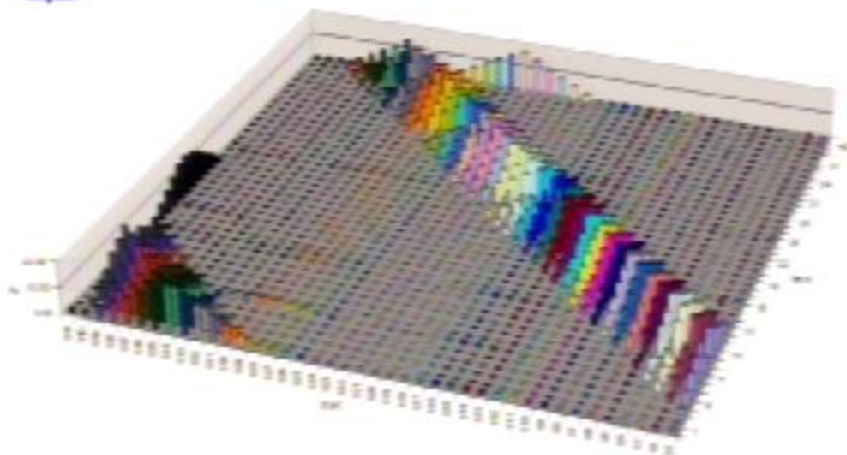


- Starting from square 1 – where do we land?



# Our Markov Matrix

---





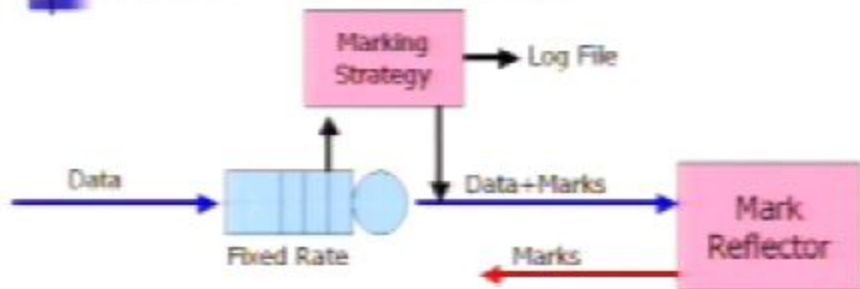


## Why learning

---

- Our example had rules, but maybe:
  - We don't know the rules
  - We can't work out the rules
  - There are too many rules
- Examples:
  - Internet Router configurations
  - System log files

## ECN



- ECN -- Explicit Congestion Notification
- Data is marked according to some strategy
  - Current RFC currently uses RED algorithm



## Congestion pricing

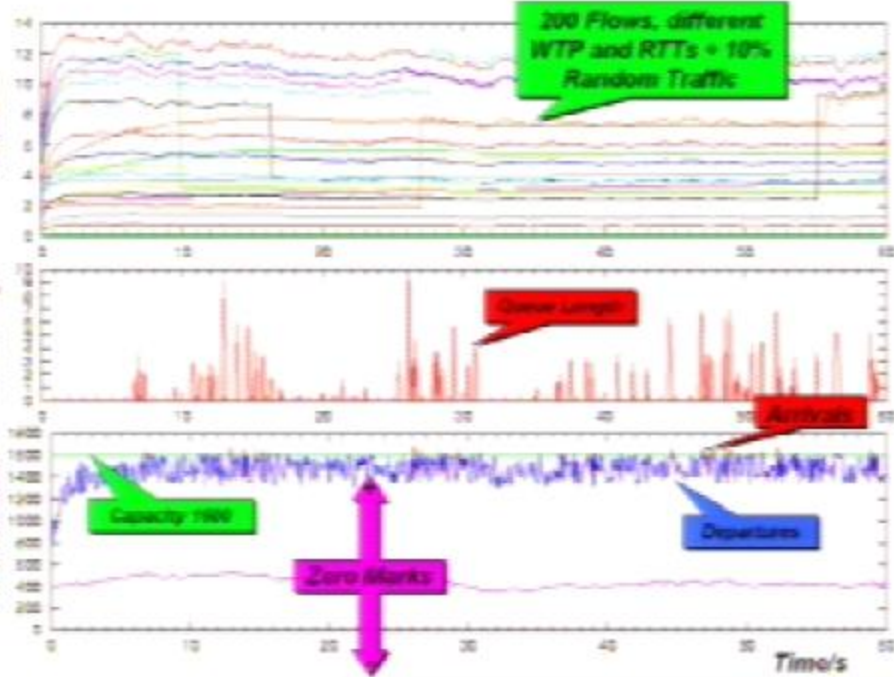
---

- Economic base strategy
- Original proposal for 1 bit marks
- Current work using 8 bit marks
  - Marks range from -128 to 127
  - Zero marks at (capacity - headroom)
  - Headroom adjusted by
    - Queue length
    - History
  - WTP is per packet target mark value

Queue Length

Total Packets/s

Packets/s





## Online measurement

---

- Large deviations
  - Interested in tails of distribution
  - Traffic unknown, multiplex complex
  - Measure....
- SVM
  - Synchronization and periodic effects
  - Try to predict behaviours
- Stability and closed loop effects?



## FCOM+

---

- Validator and Verifier for IL Assembly Code
- Modelling the COM+ Runtime Verification and Execution algorithms
- Testing both the COM+ Runtime C++ Verifier and our own, including coverage information
- Specifying and Verifying type soundness for the model



## C# generics

---

- High-level design work
- Detailed specs
- A new C# compiler
- Changes to the runtime (and IL)
- Accepted for v.2 (nearly got into v.1)

[http://camweb/Extended\\_IL/Generics/index.htm](http://camweb/Extended_IL/Generics/index.htm)



## A comment

---

<http://windows.oreilly.com/cgi-bin/comments>

"I also hope they talk to someone who's heard of ML or Haskell before they add generics to the language."

- ML compiler for .NET
- Haskell compiler for .NET





## A comment

---

<http://windows.oreilly.com/cgi-bin/comments>

"I also hope they talk to someone who's heard of ML or Haskell before they add generics to the language."

- ML compiler for .NET
- Haskell compiler for .NET



# What happens if....

Financial engineering

Programming  
language design and  
implementation

Jean Marc



Simon  
and  
Julian



# You have a language problem

Swaps, caps, options,  
european, bermudan,  
straddle, floors, swaptions,  
swallows, spreads, ...

Functions, types,  
lazy evaluation, modularity,  
composition, Haskell, ...

Jean Marc



Simon  
and  
Julian



## American options

```
handcuff :: Date -> Date -> Date -> Float -> Float -> Contract
handcuff t1 t2 t3 n price
  = after t1 (anytime (truncate t2 (shares 'or' zero shares))
  where
    shares = quote t3 n M$Shares 'and'
            give (quote t3 price Dollars)
```

Extensible  
library

Combinators



## So what....

```
give      :: Contract -> Contract
or        :: Contract -> Contract -> Contract
and       :: Contract -> Contract -> Contract
zero     :: Contract -> Contract
anytime  :: Contract -> Contract
...and some more besides...
```

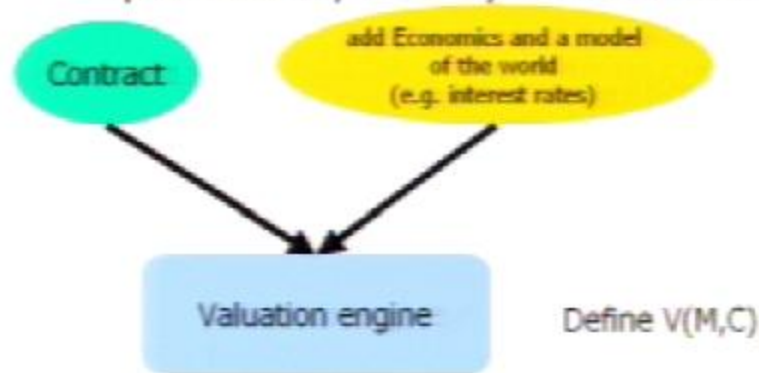
- Choice of combinators driven by
  - Economy (as few as possible)
  - Expressiveness (can describe many contracts)
  - Efficiency (maps cleanly onto e.g. valuation engine)



# Valuation

---

- Once we have a precise contract specification, we may want to value it





## ...big win

$V(M,C)$  is compositional

Add value trees  
point-wise

$V(M, c1 \text{ 'and'} c2) = V(M,c1) + V(M,c2)$

$V(M, c1 \text{ 'or'} c2) = \max( V(M,c1), V(M,c2) )$

$V(M, \text{give } c) = - V(M,c)$

$V(M, \text{anytime } c) = \text{snell}( V(M,c) )$

$V(M, \text{get } t \text{ } c) = \text{discount}( V(M,c)[t] )$

...etc...





## And finally

---

- <http://msrweb>
- <http://camweb/fridaytalk/>